Міністерство освіти та науки України

Інститут «Телекомунікації, радіоелектроніки та радіотехніки»

Изображение выглядит как текст, круг, Шрифт, эмблема

Автоматически созданное описание

Звіт

З лабораторної роботи №8

З дисципліни «Об`єктно-орієнтоване програмування»

На тему: «Конструювання базових та похідних класів»

Виконав:

Студент групи АП-22

Іщак Д.А.

Прийняв:

Алтунін С. І

Львів 2024

**Мета:** Створити, відлагодити та протестувати програму, у якій створити базовий та похідні класи для обробки даних.

**Хід роботи:**

1. продумайте, які типи змінних вам потрібно використати для виконання завдання.

2. створіть проєкт із кількома файлами, тобто крім основного файлу, наприклад main.cpp, додайте для кожного класу заголовний файл <class>.h та виконуваний файл <class>.cpp. Також модифікуйте ..\.vscode\tasks.json, щоб вказати йому шлях до всіх файлів проекту.

3. у заголовних файлах оголосіть базовий (батьківський) та похідний (дочірній) класи, а також відповідні структури для збереження і обробки даних. Для початку можете спробувати розробити програму із publicчленами оголошених класів, а потім модифікуйте її так, щоб члени батьківського та дочірнього класів були private.

4. у виконуваних файлах реалізуйте конструктори та деструктори, методи для введення і виведення даних.

5. методи введення чи зміни полів класу повинні перевіряти валідність заданих змінних і попереджувати користувача про некоректні значення.

6. можете збільшити кількість членів класу додатковими полями чи методами.

7. обчислити результат відповідно заданого варіанту, наприклад загальну вартість, тривалість, кількість, максимальне чи мінімальне значення тощо.

8. програма має вивести на консоль (термінал) вхідні дані та результат виконання.

**Варіант завдання:**



**Код програми:**

**trip.h:**

**#ifndef TRIP\_H**

**#define TRIP\_H**

**#include <iostream>**

**#include <string>**

**#include <limits>**

**// Базовий клас**

**class Product {**

**protected:**

**std::string name; // Назва товару**

**int quantity; // Кількість товару**

**public:**

**Product(const std::string& name = "", int quantity = 0)**

**: name(name), quantity(quantity) {}**

**// Методи для вводу даних**

**virtual void input() {**

**std::cout << "Введіть назву товару: ";**

**std::cin >> name;**

**while (true) {**

**std::cout << "Введіть кількість товару: ";**

**std::cin >> quantity;**

**if (std::cin.fail() || quantity < 0) {**

**std::cout << "Некоректне значення. Спробуйте ще раз.\n";**

**std::cin.clear();**

**std::cin.ignore(std::numeric\_limits<std::streamsize>::max(), '\n');**

**}**

**else break;**

**}**

**}**

**// Метод для виводу даних**

**virtual void display() const {**

**std::cout << "Назва: " << name << ", Кількість: " << quantity;**

**}**

**};**

**// Дочірній клас**

**class PricedProduct : public Product {**

**private:**

**struct Details { // Вкладена структура**

**double price; // Ціна за одиницю**

**} details;**

**public:**

**PricedProduct(const std::string& name = "", int quantity = 0, double price = 0.0)**

**: Product(name, quantity) {**

**details.price = price;**

**}**

**// Перевизначений метод для вводу даних**

**void input() override {**

**Product::input(); // Викликаємо метод базового класу**

**while (true) {**

**std::cout << "Введіть ціну за одиницю товару: ";**

**std::cin >> details.price;**

**if (std::cin.fail() || details.price < 0) {**

**std::cout << "Некоректне значення. Спробуйте ще раз.\n";**

**std::cin.clear();**

**std::cin.ignore(std::numeric\_limits<std::streamsize>::max(), '\n');**

**}**

**else break;**

**}**

**}**

**// Перевизначений метод для виводу даних**

**void display() const override {**

**Product::display(); // Викликаємо метод базового класу**

**std::cout << ", Ціна за одиницю: " << details.price**

**<< ", Загальна вартість: " << calculateTotal() << '\n';**

**}**

**// Метод для обчислення загальної вартості**

**double calculateTotal() const {**

**return quantity \* details.price;**

**}**

**};**

**int main() {**

**// Створення об'єкта дочірнього класу**

**PricedProduct product;**

**std::cout << "Введіть дані про товар:\n";**

**product.input(); // Введення даних користувачем**

**std::cout << "\nІнформація про товар:\n";**

**product.display(); // Вивід даних на екран**

**return 0;**

**}**

**Результати виконання роботи:**

**Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание**

**Блок-схема програми:**

**Изображение выглядит как текст, снимок экрана, Шрифт, чек

Автоматически созданное описание**

Висновок:

У цій лабораторній роботі реалізовано базовий і похідні класи для обробки даних про подорожі. Програма дозволяє використовувати спадкування для зручного управління різними типами подорожей, спрощуючи розширення функціональності. Отримані знання про спадкування, віртуальні методи та управління пам’яттю дозволяють будувати складні програми із високим ступенем модульності та повторного використання коду.